# EXPERT SYSTEM VERIFICATION AND VALIDATION STUDY
## Phase 2: Requirements Identification

## Delivery 2 - Current Requirements Applicability

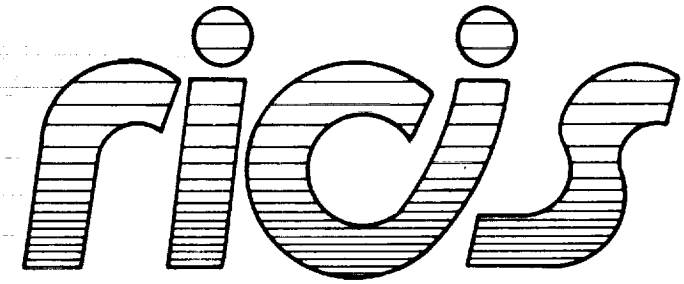


*International Business Machines Corporation*

**January 31, 1991**

**Cooperative Agreement NCC 9-16**
**Research Activity No. AI.16**

**NASA Johnson Space Center**
**Information Systems Directorate**
**Information Technology Division**



*Research Institute for Computing and Information Systems*
*University of Houston - Clear Lake*

**T·E·C·H·N·I·C·A·L  R·E·P·O·R·T**

# *The RICIS Concept*

The University of Houston-Clear Lake established the Research Institute for Computing and Information systems in 1986 to encourage NASA Johnson Space Center and local industry to actively support research in the computing and information sciences. As part of this endeavor, UH-Clear Lake proposed a partnership with JSC to jointly define and manage an integrated program of research in advanced data processing technology needed for JSC's main missions, including administrative, engineering and science responsibilities. JSC agreed and entered into a three-year cooperative agreement with UH-Clear Lake beginning in May, 1986, to jointly plan and execute such research through RICIS. Additionally, under Cooperative Agreement NCC 9-16, computing and educational facilities are shared by the two institutions to conduct the research.

The mission of RICIS is to conduct, coordinate and disseminate research on computing and information systems among researchers, sponsors and users from UH-Clear Lake, NASA/JSC, and other research organizations. Within UH-Clear Lake, the mission is being implemented through interdisciplinary involvement of faculty and students from each of the four schools: Business, Education, Human Sciences and Humanities, and Natural and Applied Sciences.

Other research organizations are involved via the "gateway" concept. UH-Clear Lake establishes relationships with other universities and research organizations, having common research interests, to provide additional sources of expertise to conduct needed research.

A major role of RICIS is to find the best match of sponsors, researchers and research objectives to advance knowledge in the computing and information sciences. Working jointly with NASA/JSC, RICIS advises on research needs, recommends principals for conducting the research, provides technical and administrative support to coordinate the research, and integrates technical results into the cooperative goals of UH-Clear Lake and NASA/JSC.

# EXPERT SYSTEM VERIFICATION AND VALIDATION STUDY

## Phase 2: Requirements Identification

## Delivery 2 - Current Requirements Applicability

## Preface

This research was conducted under auspices of the Research Institute for Computing and Information Systems by the International Business Machines Corporation. Dr. Terry Feagin and Dr. T. F. Leibfried served as RICIS research representatives.

Funding has been provided by Information Technology Division, Information Systems Directorate, NASA/JSC through Cooperative Agreement NCC 9-16 between NASA Johnson Space Center and the University of Houston-Clear Lake. The NASA technical monitor for this activity was Chris Culbert, of the Software Technology Branch, Information Technology Division, Information Technology Directorate, NASA/JSC.

# Expert System Verification and Validation Study
# RICIS Contract #069

## Phase 2 - Requirements Identification
## Delivery 2 - Current Requirements Applicability

January 31, 1991

IBM

3700 Bay Area Blvd.
Houston, Tx. 77058

**ES V&V Requirements Applicability**

# Preface

This document constitutes the second delivery, "Current Requirements Applicability," of the four deliveries scheduled for the second phase of RICIS contract #069, "Verification and Validation of Expert Systems Study."

The four deliverables in this phase are:

1. Updated Survey Report
2. Current Requirements Applicability
3. Draft Modifications
4. Final Report

# Contents

# Background

This is the second phase of a task which has the ultimate purpose of ensuring that adequate Expert Systems Verification & Validation tools and techniques are available for Space Station Freedom Program (SSFP) Knowledge Based Systems development[1]. The purpose of this phase is to recommend modifications to current software Verification & Validation (V&V) requirements which will extend the applicability of the requirements to NASA Expert Systems (ESs).

[1] In this report, the terms *Expert System (ES)* and *Knowledge-Based System (KBS)* are considered as synonymous terms and are used interchangeably.

# Executive Summary

Although Knowledge Based Systems (KBSs) are expected to be heavily used in the Space Station Freedom Program (SSFP), no work has been dedicated to developing tools or techniques which are specifically targeted at supporting the KBS verification and validation (V&V) needs of SSFP. Sufficient differences, in both the approach to V&V and the execution of V&V, prevent the application of many conventional software V&V tools and techniques to KBSs. In addition, few KBS V&V approaches have been applied to operational systems.

## Results From Previous Phase

**Phase I** Established the state-of-the-practice relative to the Verification and Validation of KBSs. This was accomplished through a survey and interview involving approximately 70 individuals experienced in KBS research and/or development.

## Phase II Objective

This phase addresses the specific problem of a lack of understanding about SSFP KBS V&V needs. Although V&V requirements have been identified for major portions of the SSFP, it is not directly known how these requirements affect KBSs. Some requirements may be directly applicable, while others may only indirectly apply (i.e., the requirement, as stated, does not apply but the "spirit," or intent, of the requirement does apply). Also, there may be unique requirements for KBSs not covered in the current set of requirements.

### Deliverable Objective and Results

This deliverable, "Current Requirements Applicability," determines the applicability of the current SSFP conventional software V&V requirements to V&V of KBSs. This was accomplished through a series of tasks:

1. Identification of SSFP conventional software V&V requirements
2. Analysis of conventional requirements for KBS relevance
3. Comparison of conventional requirements to Phase I survey results

Thirty-four of the 50 requirements isolated in this deliverable can be satisfied without any further research. Of the remaining requirements, nine can be satisfied through changes in the life-cycle (four) or requirements definition (five) process, but there currently is no process for applying seven of the requirements.

## Goals of Subsequent Phases

**Phase III** Evaluate KBS V&V approaches for applicability to SSFP
**Phase IV** Develop tools, techniques, and guidelines
**Phase V** Test tools and techniques on appropriate NASA applications
**Phase VI** Install tools in relevant SSFP environments

# Research Approach

The strategy for determining the modifications to current software V&V requirements consists of four steps. The first of these steps, which is the subject of this deliverable, is:

**Analyze current conventional software V&V requirements for applicability to KBSs.**
This activity consists of the following steps which are detailed in "Deliverable Approach":

1. Select a set of conventional software V&V requirements
2. Analyze the current requirements for V&V relevance
3. Compare the requirements to the Task 1 survey results

The remaining steps to determine the modifications to current software V&V requirements, which will be performed after this deliverable, are:

**Develop draft recommended modifications**
This activity will result in a complete list of ES V&V requirements. V&V requirements that do not directly apply to ESs, but whose intent applies, will be modified so that they are appropriate for ESs. Additional requirements will also be defined.

**Solicit review comments**
To ensure acceptance of the KBS V&V requirements by KBS developers, it will be important to solicit review comments from KBS developers to ensure that:

- All their concerns are addressed
- The V&V requirements are described in a manner that developers can easily understand
- They are aware of the KBS V&V requirements before they are formally proposed.

Reviews by developers will be ensure the completeness and accuracy of the requirements.

**Finalize recommended modifications**
Review comments will be incorporated into the draft recommended modifications to produce the final recommended modifications.

# Deliverable Approach

The first step in the research approach for this phase of the task, provided by this delivery, is to analyze current conventional software V&V requirements for applicability to KBSs. This research step consists of the following activities:

- Selection of Conventional Software V&V Requirements
- Categorization of Requirements
- Analysis of Requirements for V&V Relevance
- Comparison of Requirements to Task 1 Survey Results

The result of this analysis is contained in a set of tables, provided in "Requirements Analysis Tables" on page 10. These tables are organized by the requirement categorization described in "Categorization of Requirements" on page 4.

## Selection of Conventional Software V&V Requirements

Since the ultimate goal of this research is to ensure that adequate ES V&V tools and techniques are available for SSFP, the SSFP software V&V requirements were selected. The SSFP documents which were relevant to this research included:

- SSP 30000, Section 9, "Product Assurance Requirements"
- SSP 30467, Vol. 1, "Master Verification Requirements"

Several additional documents were reviewed for relevance to this research but did not contain the expected information. Among these documents were:

- DR SY-03.1 (WP-2) "Software Management Plan"
- LMSC F255442, "SSE Systems Methods Manual"
- DSTL-90-006, "Expert System Development Reference Manual"

## Categorization of Requirements

Each of the applicable requirements was placed in one of the following categories:

**Items That Must Be Explicitly Identified and Maintained**
These are items that are ***inputs*** to the V&V process. That is, these are items that must be specified before V&V can be done. As the systems change, these items must be updated and maintained so that V&V of the updated system can be performed.

**Attributes of Work Products That Must Be Verified, Analyzed or Tested**
There are many desirable properties or attributes of software. Obviously, it is desirable that software work correctly but it should also desirable that software be built so that it is easy to maintain, it is safe, it is easy to use, etc.. It is necessary to explicit state the important desirable properties so they can be verified.

**Items That Must/Do Not Need to Undergo V&V**
Software can be come from several sources (development engineer, payload specialist, or a commercial product) and be used for different purposes (onboard operations, development support, etc.). Based on its source and purpose, an item of software may not need to undergo V&V.

**V&V Requirements Relating to the V&V Process**
These requirements deal with how the V&V process is to be managed and controlled.

**Methods of V&V**
These requirements relate to ***how*** V&V can be done.

**Miscellaneous V&V Requirements**
These are requirements that do not fall into one of the main categories.

## Analysis of Requirements for V&V Relevance

This requirements analysis was determined from results of the Phase I survey, other *state-of-the-art* and *state-of-the-practice* sources, knowledge of ES V&V issues, experience in developing ESs, and general knowledge of ES technology. The list of requirements was decomposed into four sublists:

1. Requirements which do not apply to KBSs.

   This list is defined by omission (the requirements are not included in the "Requirements Analysis Tables" on page 10). The following types of requirements have been omitted from consideration:

   - V&V requirements which do not apply to software
   - Logistics requirements
   - Documentation requirements
   - Reporting procedures

(The following sublists are provided in "Requirements Organized by Problem Status" on page 8.)

2. Requirements which can be applied equally to KBSs and conventional software systems.
3. Requirements which can be applied to KBSs using existing procedures.
4. Requirements which are not satisfied by the state-of-the-practice, but could be satisfied using existing processes.
5. Requirements for which the intent applies to KBSs, but there is no existing process.

For the list of requirements which were analyzed, there were no requirements relevant to conventional software systems which do not apply to Expert Systems, even in intent.

### Comparison of Requirements to Task 1 Survey Results

Phase I of this task produced an understanding of the current state of the practice in V&V of ESs. These survey results were compared to the V&V requirements in order to determine:

- Which V&V requirements are currently being satisfied.
- Why certain V&V requirements are not being satisfied (possibly they do not apply or are difficult to satisfy).

## Research Notes

There are three major topics upon which the remaining research is based. The first of these topics concerns the differences between conventional software and knowledge based systems: an understanding of the differences is essential for analyzing V&V requirements. The second topic concerns the software life-cycle model used in development. While the SSFP requirements do not dictate a specific software life-cycle, the requirements are specified in a way that assumes that the waterfall type of software life-cycle will be used in development of all SSFP software. This assumption has significant implications on SSFP KBS development. Finally, the SSFP requirements specify all the types of requirements which must be developed and imply the level of detail to which they must developed. These requirements can not be easily met following the state-of-the-practice in KBS development. Each of these topics and their impacts is discussed in the following sections.

In "Requirements Analysis Tables" on page 10, the V&V requirements which are affected by the process conflict are indicated by the *Issue (Life-Cycle)* Problem Status and the V&V requirements which are affected by the requirements issue are indicated by the *Issue (Requirements)* Problem Status. (The total set of Problem Status values is provided in "Requirements Organized by Problem Status" on page 8.)

### Differences Between Conventional Software and Knowledge Based Systems

An understanding of the differences between KBSs and conventional software is key to analyzing the relevance of current SSFP V&V requirements to KBSs. We will define a KBS to be a system that solves a problem in the same manner as a human expert, using a representation of the expert's knowledge and possibly using the same problem solving method as the expert. (Note that we do not mean that a KBS must be implemented in some type of "AI language.") A conventional software system is one that solves a problem using conventional algorithms and data structures. An important point of these definitions is that the difference between a KBS and a conventional software system lies in ***how*** the problem is solved, not in ***what*** problem is solved. Some types of problems appear to be better suited to be solved with a KBS while others appear to be better suited to be solved with a conventional software system. However, we will not take a system to be a KBS simply because it solves a certain type of problem; it must solve it in a particular way.

If we had defined a KBS in terms of the type of problem that it solves, we would have had the following problem. KBSs are often developed for problems that are inherently very difficult; they are often problems for which conventional software approaches have failed to solve. If we had defined a KBS in terms of the problem it solves, we would have been forced to conclude that KBSs are inherently very difficult to V&V simply because their problems are very difficult. Instead, by focussing on how KBSs solve a problem and how KBSs are developed, we can better identify V&V differences.

## Space Station Requirements Life Cycle Bias

Although there is no requirement to use a specific life-cycle model for software V&V, many of the V&V requirements implied a type of life-cycle model. For example, in SSP 30467 (Master Verification Requirements, Vol. 1, hereafter referred to as "the MVR"), it is stated that "System certification shall be based on verification of design requirements applicable to each level of hardware and/or software development." This implies that the development and verification process is divided into a sequence of "levels" as in a typical waterfall life-cycle model. Section 3 of the MVR (a "for information only" section) somewhat clarifies this implication by discussing a standard verification process where the term "level" is used to refer to levels of integration (unit, subsystem, and system). Based on the discussion in section 3, it is reasonable to assume that "levels of design" probably refers to unit design, subsystem design, and system design. This example shows that, although the specific process described in section 3 is not a requirement, an understanding of the general process, or life-cycle model, is needed in order to correctly interpret many of the requirements.

There is a conflict between the preferred KBS development life-cycle, or process, and the one implied by the current SSFP requirements. The preferred KBS process typically begins with a very ill-defined set of requirements which are refined in parallel with development in a highly iterative fashion (the Spiral Model[2], which is a cyclic model, is often used as a base for defining a KBS life-cycle model). In contrast, the implied SSFP process is one where a complete understanding of the system is stated, analyzed, and then refined into a lower "level" design. The primary conflict between these two different processes is the amount of freedom allowed to change the system; the KBS process allows a large amount of freedom while the implied SSFP process greatly restricts the freedom. The MVR appears to recognize this lack of freedom in the SSFP process because the MVR specifically discusses a separate development activity, called software prototyping, which allows for *complete* freedom to change this system; that is, there are essentially *no* V&V requirements associated with software prototyping. This is not the same as the KBS process where prototyping is not a separate activity but instead is integrated into the process.

Because of this conflict, several of the SSFP V&V requirements are difficult to satisfy. That is, they can be satisfied but only at great expense. For example, SSP 30000, section 9 (Product Assurance Requirements), paragraph 4.2.3 states that "Engineering changes shall be reviewed by Quality Assurance to determine the quality impact ... "; i.e., each change to the requirements must be approved by a Quality Assurance organization. Also, the change process as described in SSP 30000, section 2, part 9 (Configuration Management Requirements) requires significant documentation to implement and release changes. This imposes a large overhead in order to accommodate rapidly changing requirements.

## State-of-the-Practice Requirements Detail

There are many types of requirements that are necessary to support V&V. The types of system requirements[3] indicated in the V&V requirements are external, operational, functional, performance, interface, support, design, reliability, safety, maintainability, quality, and certification requirements. In order to satisfy the V&V requirements, all of these types of system requirements must be specified. There is no reason to believe that it is inherently impossible to state all of these requirements for any KBS. However, the current state-of-the-practice of KBS development does not involve writing these types of requirements. And there is a wide belief that stating many of these types of requirements for some KBSs is too difficult to be practical.

---

[2] R.M. O'Keefe, L. Sunro, "An Integrative Model of Expert Systems Verification and Validation," *Expert Systems with Applications*, (1990).

[3] There are requirements on the development process (e.g., the V&V requirements discussed here) as well as requirements on the system being developed. We will refer to the former type of requirements as "process requirements" and the latter type as "system requirements."

It is not clear from the V&V requirements whether system requirements must be specified to a certain level of detail. However, the MVR states in paragraph 4.2.1.AI, "Software verification requirements shall ensure that all SSP software is tested to a set of uniform requirements ...." We interpret "uniform requirements" to mean a set of requirements that can be analyzed to be consistent. It will not be possible to perform this consistency analysis if some parts of the requirements are not sufficiently detailed (i.e., If some one requirement is much less detailed than other requirement, the two requirements many be inconsistent but there is insufficient detail to determine this.) So we must conclude that system requirements for a KBS can not be written at a very high level in order to easily satisfy the V&V requirements.

The issue of stating all types of system requirements for a KBS is probably the most important general KBS V&V issue. This is because there are many V&V requirements that depend on the existence of system requirements. It is also appears to be one of the most difficult issues to resolve. There is a wide difference between the state-of-the-practice, in stating KBS system requirements, and what will be needed to satisfy SSFP V&V requirements. Convincing KBS developers to improve their current practice may be very difficult since it is widely believed that it is not practical to state system requirements for a KBS. Finally, there is no clear method of generating KBS system requirements; there does not even appear to be a good example of what KBS system requirements should be like.

# Requirements Organized by Problem Status

Each requirement which is relevant to KBS V&V, as detailed in "Requirements Analysis Tables" on page 10, is categorized according to "Problem Status." Problem Status defines whether the requirement presents a problem for which there is currently no solution, is currently a problem in the state-of-the-practice (either because of life-cycle or requirements issues), is not a problem in current practice, or is applied to conventional software and KBSs in the same manner.

This section organizes the requirements by the Problem Status and provides the definition of each status value. The page on which the requirement details can be found is included with each requirement.

**Problem** The intent of the requirement applies to KBSs, but currently there is no process for applying the requirement.

- Components (Parts) on page 11
- Maintainability on page 14
- Requirements to Code on page 15
- Performance (Resource Usage) on page 16
- Off-the-shelf (OTS) Software Components on page 17
- Paths (Redundant Paths, Decision Paths, Executable Lines of Code) on page 17
- IV&V Must Be Performed (At Least On Critical S/W) on page 19

**Issue (Life-Cycle)**
The requirement is a problem in the state-of-the-practice. The definition of the life-cycle process will be determined in part by the manner in which this requirement is met. Those requirements which can be determined once the life-cycle is defined are not classified in this category.

- Temporary Systems on page 11
- Development Methods (to ensure they support Q&A) on page 15
- Q&A Must Approve Proposed Changes on page 18
- CM Must Be Done on page 18

**Issue (Requirements)**
The requirement is a problem in the state-of-the-practice. The definition of the requirements for KBSs will be determined in part by the manner in which this requirement is met. Those requirements which can be determined once the requirements are defined are not classified in this category.

- Test Pass/Fail Criteria on page 12
- Requirements Identification on page 12
- Quality on page 14
- Verification of Requirements on page 15
- Analysis Methods Must Be Based on Sound Engineering Approaches on page 20

**Not a problem** The requirement can be applied to KBSs using existing processes. This category also contains those requirements which depend on the life-cycle or requirements definition, but are not affected by the details of the definition.

- Product Assurance Tasks on page 11
- V&V Training on page 11
- Levels of Verification on page 12
- Reliability on page 14
- Faults/Failures (Analyze Their Impact) on page 15
- V&V Must Be Done Throughout the Life Cycle on page 18
- Development Testing (Informal Testing Instead of Analysis) on page 19

**Not unique** The requirement is defined either at a level of generality or at a point in the life-cycle where specific software attributes are indistinguishable, and can be applied equally to both KBSs and conventional software systems. There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.

# Requirements Analysis Tables

These tables contain the both the analysis of the requirements for KBS V&V relevance and the comparison to the results of the Task I survey results. The following information is provided by these tables:

**V&V Item** A short descriptive name for the V&V topic.

**Problem Status** The problem status has one of the values as specified in "Requirements Organized by Problem Status" on page 8.

**Paragraph** Each requirement is referenced by SSP document and paragraph number. The SSP document abbreviations are as follows:

**PA** SSP 30000, Section 9, "Product Assurance Requirements"
**MV** SSP 20467, Vol. 1, "Master Verification Requirements"

**V&V Requirement** The major topic (often paraphrased) in the referenced paragraph. The V&V requirement may or may not be directly quoted from the documents.

**Analysis and Comparison**
Contains both the results of comparing the requirements to the survey results, and the analysis of the relevance of the requirements to KBSs.

Each of the KBS V&V requirements appears in one of the following tables, which correspond to the categories of requirements as defined in "Categorization of Requirements" on page 4.

- Items That Must Be Explicitly Identified and Maintained (Table 1 on page 11)
- Attributes of Work Products That Must Be Verified, Analyzed, or Tested (Table 2 on page 14)
- Items That Must Undergo V&V (Table 3 on page 17)
- V&V Requirements Relating to the V&V Process (Table 4 on page 18)
- Methods of V&V (Table 5 on page 19)
- Miscellaneous V&V Requirements (Table 6 on page 20)

Table 1 (Page 1 of 3). Items That Must Be Explicitly Identified and Maintained

| V&V Item / Problem Status | Paragraph / V&V Requirement / Analysis and Comparison |
|---|---|
| Product Assurance Tasks<br>*Not a problem* | **PA 1.3.A** Product assurance tasks must be defined.<br>Product assurance tasks are defined based on the life cycle model in use. The definition of these tasks should not be a problem once a life cycle model is selected.<br>The results of the survey indicate that many projects use no life cycle model while others seem to use an informal one. This infers that the identification of product assurance tasks is probably not currently done. |
| Potential Hazards<br>*Not unique* | **PA 2.2.1.A** Potential hazards should be identified and addressed.<br>**MV 4.2.1.N** All criticality 1 hazards shall include some type of warning device, procedures for what to do when they occur, and/or a method for containing them.<br>Identification of potential hazards (or "disasters") has been cited as an important and useful method for knowledge acquisition. Identification of potential hazards for KBSs is no different than for conventional software systems.<br>The survey did not include a question on the identification of potential hazards. A question on the extent of the worst possible hazard indicated that identifying hazards was not a problem. Interviews, in which interviewees readily discussed potential hazards, reinforced these findings. |
| Criticality<br>*Not unique* | **PA 3.2.4** This paragraph defines the criticality categories to be used.<br>**PA 5.5.7** All software shall be analyzed to determine if it is critical.<br>Criticality is currently (or can be) identified for KBSs.<br>The survey contained a multiple-choice question establishing the criticality of the KBS. Since almost everyone responded to this question, each of the loosely categorized criticality levels was selected at least once, and no one selected the "other" category, it seems that Criticality is currently (or can be) identified for KBSs. |
| V&V Training<br>*Not a problem* | **PA 4.1.6** The need for training of Q&A personnel must be addressed.<br>Courses for V&V of Expert Systems currently exist. |
| Components (Parts)<br>***Problem*** | **PA 4.3.1** Each article should be identified by a unique part number.<br>The letter of this requirement would allow identification by KBname.rulename; however, the intent of this requirement is to allow management of meaningful chunks (modules) of information. Modularity is a known problem in KBS research for which there is not a good state-of-the-art method.<br>While the survey did not directly indicate that modularity was a problem, analysis of a collection of the questions revealed that modularity, along with readability and solution complexity, was a problem. |
| Temporary Systems<br>*Issue (Life-Cycle)* | **PA 4.5.7** Temporary systems (prototypes) must be identified and logged.<br>In many KBS development efforts, successive prototypes are developed until the final prototype is deemed the production system. Other systems which are meant to be prototypes (temporary systems) are placed in production. Definition of a formal life-cycle should include a way to determine whether the system will be temporary.<br>The survey indicated that there probably is some difficulty identifying temporary systems since there seemed to be some difficulty distinguishing prototypes from operational systems. |
| Test Procedures, Data, and Plans<br>*Not Unique* | **MV 4.2.9.B** V&V activities shall include: Production of test plans, requirements and procedures.<br>**PA 4.6.2** Test procedures must be documented and maintained for later review.<br>**PA 4.6.3.C** Test data and results should be maintained or reproducible.<br>**MV 4.2.1.C** Test data shall be made available to anybody in SSFP.<br>**MV 4.2.1.B** Test plans are required and required to be tailored to the specific operational requirements.<br>The frequency of updates in current KBS development efforts makes management of test procedures, data, and plans more difficult, but otherwise there is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey did not address this concern. |

Table 1 (Page 2 of 3). Items That Must Be Explicitly Identified and Maintained

| V&V Item<br>Problem Status | Paragraph V&V Requirement<br>Analysis and Comparison |
|---|---|
| Test Pass/Fail Criteria<br>*Issue (Requirements)* | **MV 4.2.1.L** Each test must have pass/fail criteria.<br>Many KBSs are designed to provide approximate answers, complicating the definition of pass/fail criteria. It is probably true that pass/fail criteria is often difficult to identify (and may be the root problem behind defining requirements).<br>It is not clear from the survey results whether pass/fail criteria is routinely identified. Survey responses indicate comparatively low expected and actual accuracy for KBSs, suggesting that complete definition of pass/fail criteria may be difficult for some systems. 10% of the survey respondents did not know how often the system was correct. |
| Software Approved/ Certified for a Particular Use<br>*Not unique* | **PA 5.2.5** Consistency between software used and what should be used shall be maintained (i.e., CM shall be done).<br>**PA 5.10** This paragraph defines certification.<br>The application of this requirement does not differ between KBSs and conventional software systems.<br>From some of the interview results, it does appear that this was done in some cases. |
| Requirements Identification<br>*Issue (Requirements)* | **PA 5.3.3** Though not explicit, it is implicitly stated in several places that requirements shall be written and maintained.<br>**MV 4.2.3.A** Testing shall cover:<br>a) Design requirements<br>b) External, operational, functional, performance, interface, and support requirements<br>c) Requirements compliance at several points in the life cycle.<br>Requirements identification is currently a problem only when it is not performed. It is common in KBS development to incorporate requirements changes immediately into the KBS, without otherwise documenting the requirements. Stating detailed requirements of all the types required is likely to be difficult, as discussed in "State-of-the-Practice Requirements Detail" on page 6.<br>Less than half of the survey participants indicated that any written requirements were available and less than a third indicated that a requirements document was written as part of development. Those participants that had written requirements available did not indicate that there was any difficulty in generating the requirements, though it is unlikely that generated all the required types. |
| Levels of Verification<br>*Not a problem* | **MV 4.2.1.AJ** Where and when each level of verification is to be performed shall be identified.<br>The definition of levels will depend on the identification of a KBS module, which, as described by Components (Parts) on page 11, is a problem. Given a definition of a module, however, determination of the levels of verification should not a problem. Once a life-cycle model is selected, determination of when to perform verification also should not be a problem.<br>The survey did not contain direct questions on levels of verification, but did establish that in most cases evaluation was performed by the developer, then the expert, and then the user. |
| External and Support Requirements<br>*Not unique* | **MV 4.2.3.A** Testing shall cover: External, operational, functional, performance, interface, and support requirements.<br>It is not clear what is meant by "external" and "support" requirements, but it does not seem that there is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey did not address this concern. |
| Certification Requirements<br>*Not unique* | **MV 4.4.2.A** Certification requirements shall be defined individually for each component.<br>Other than the need for a KBS definition of components, there is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey did not question certification requirements, but some interview results that indicated that certification requirements existed (flight controller certification tests). |

| Table 1 (Page 3 of 3). Items That Must Be Explicitly Identified and Maintained | |
|---|---|
| **V&V Item**<br>**Problem Status** | **Paragraph V&V Requirement**<br>**Analysis and Comparison** |
| Certification Level<br>*Not unique* | **MV 4.4.2.E** Certification level required is based on:<br>• Impact of component on safety<br>• Potential impact of errors on the mission success<br>• Technical complexity<br>• System classification<br>• System size<br>• Cost and schedule impact of certification |
| | There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey results only indicated that some KBSs were certified. |
| Endurance Test Time<br>*Not unique* | **MV 4.5.N** Software endurance testing shall be performed on all software during V&V, the time duration is determined on a case-by-case basis. |
| | Selecting a meaningful time duration, while potentially difficult, should not differ from selecting a time duration for endurance testing of conventional software systems.<br>Endurance testing was not covered either by the survey or by the interviews. |

Table 2 (Page 1 of 3). Attributes of Work Products That Must Be Verified, Analyzed or Tested

| V&V Item<br>Problem Status | Paragraph | V&V Requirement |
|---|---|---|
| | **Analysis and Comparison** | |
| Safety<br>*Not unique* | PA 1.3.B | Safety, reliability, maintainability, and quality must be evaluated. |
| | PA 2.1.1.A | All hazards must be identified. |
| | PA 2.1.1.B | All hazards must be eliminated or controlled. |
| | PA 2.1.1.C | An overall safety risk assessment must be made. |
| | PA 2.2.3 | Analysis must include likelihood of hazard occurrence and effects. |
| | PA 5.6 | SQA shall ensure that hazard analysis is performed and that the resulting safety requirements are satisfied. |
| | Once safety is identified in the form of requirements, analyzing that a KBS satisfies safety requirements should be no different than for conventional software systems.<br>The survey included a question on what the worst affect of a system failure could be, and the responses indicated that the hazards could be identified. | |
| Reliability<br>*Not a problem* | PA 1.3.B | Safety, reliability, maintainability, and quality must be evaluated. |
| | PA 3.1.5 | When data needed for reliability and maintainability analysis is needed as GFE, it should be identified. |
| | PA 3.2.1 | Reliability and maintainability design criteria should be identified/ addressed before/ during design. |
| | PA 3.2.2.1 | Design trade-offs should be analyzed for reliability concerns as well as other criteria (e.g., performance, etc.). |
| | PA 5.5.6 | Reliability and maintainability of the architecture and design shall be analyzed. |
| | There is at least one paper on applying reliability modelling to KBSs, indicating that verifying reliability, either subjectively or objectively does not appear to be a problem.<br>The survey did question the reliability of KBSs as compared with conventional software systems, and revealed that (subjectively) KBSs had roughly the same reliability. | |
| Maintainability<br>***Problem*** | PA 1.3.B | Safety, reliability, maintainability, and quality must be evaluated. |
| | PA 3.1.5 | When data needed for reliability and maintainability analysis is needed as GFE, it should be identified. |
| | PA 3.2 | Maintainability of KBS must be evaluated. |
| | PA 3.2.1 | Reliability and maintainability design criteria should be identified/ addressed before/during design. |
| | PA 3.2.2.2 | Design trade-offs should be analyzed for maintainability concerns as well as other criteria (e.g., performance, etc.). |
| | PA 4.1.4 | Consideration and planning for both scheduled and unscheduled on-orbit maintenance must be made. |
| | PA 5.5.6 | Reliability and maintainability of the architecture and design shall be analyzed. |
| | MV 4.10 | Maintenance characteristics of software shall be progressively verified and demonstrated during design, development, test and operations. |
| | The lack of maintenance properties will cause problems for evaluation of KBS maintainability, performing design trade-offs concerning maintainability, and verifying and demonstrating maintenance characteristics. It is also unlikely that the performance of on-orbit maintenance of KBSs is well understood.<br>There was not a direct survey question on this, but other questions and interview results indicate that while there is not presently an explicit maintenance concern, no maintenance properties seem to have been established. Interviewees were asked about maintainability properties, but could not identify what these properties could be. | |
| Quality<br>*Issue (Requirements)* | PA 1.3.B | Safety, reliability, maintainability, and quality must be evaluated. |
| | The SSFP requirements do not define quality; however, quality is normally associated with errors per line of code, which does not readily apply to knowledge bases. A quality metric for KBSs (errors per rule?) needs to be selected to satisfy the intent of the requirement.<br>The survey did not directly question quality. | |
| (Proposed) Operational Use<br>*Not unique* | PA 5.1.9 | A systematic method for analyzing the operational use of S/W shall be developed. |
| | MV 4.2.3.A | Testing shall cover: External, operational, functional, performance, interface, and support requirements. |
| | There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey results showed that many KBSs were executed in parallel in an operational environment. | |

Table 2 (Page 2 of 3). Attributes of Work Products That Must Be Verified, Analyzed or Tested

| V&V Item / Problem Status | Paragraph | V&V Requirement |
|---|---|---|
| | **Analysis and Comparison** | |
| Development Methods (to ensure they support Q&A)<br>*Issue (Life-Cycle)* | PA 5.2.2<br>PA 5.2.6 | Techniques and methods for development shall be reviewed to determine if they support Q&A.<br>SQA shall review the process and suggest changes to decrease the chances of introducing nonconformities. |
| | | The KBS development process in most cases is not formal enough to suggest that formal development methods are used. Development methods (which support Q&A) will need to be defined along with a life-cycle model for KBS development.<br>Survey results indicate that development methods were informal which implies that it is unlikely that they were analyzed to see if they supported verifiability. However, in at least one case, a language was forbidden (LISP) out of concerns for verifiability (of self-modifying code). |
| Software Architecture as Well as Detailed Code<br>*Not unique* | PA 5.2.3<br>PA 5.2.4 | Software design and documentation shall be reviewed at the "architectural" and "detailed" levels.<br>Code should be inspected prior to integration and test. |
| | | Since a software architecture is basically a collection of modules and their interfaces with off-the-shelf software components, verification requires that the modules cover all the requirements and that the modules interface correctly with the off-the-shelf components and each other. Application of this requirement is dependent on the successful definition of a KBS component (see Components (Parts) on page 11), and verification of KBS interfaces (see Interfaces on page 16).<br>The survey did not contain any information on verification of the architecture. |
| Faults/Failures (Analyze Their Impact)<br>*Not a problem* | PA 5.4.2<br>PA 5.5.4 | Faults/failures discovered should be analyzed to determine their impact.<br>Failures are to be analyzed to determine error-type and do trend analysis to point out potential problems in the process. |
| | | While slightly different from determination of the impact of faults/failures in conventional software systems, it is currently possible to perform this activity on KBSs.<br>The survey did not question the difficulty of analyzing the impact of an individual fault or failure. The closest relevant question was about the sensitivity of the knowledge base to changes. Sensitivity did not appear to be a problem so it seems that analyzing the effects of a fault in the knowledge base (via debugging) would not be a problem. |
| Verification of Requirements<br>*Issue (Requirements)* | PA 5.5.3<br>MV 4.2.3.A<br><br><br><br>MV 4.2.9.B<br>MV 4.4.6 | Formal review will occur to analyze the S/W requirements derived from system requirements.<br>Testing shall cover:<br>a) Design requirements<br>b) External, operational, functional, performance, interface, and support requirements<br>c) Requirements compliance at several points in the life cycle.<br>V&V activities shall include: Analysis of requirements for correctness and testability.<br>Certification tests shall cover the full range of design requirements. |
| | | There are many issues concerning KBS requirements definition which make recording of the requirements difficult. Verification of requirements will remain a problem until the definition of requirements, which must also consider verifiability, can be performed adequately.<br>Although the survey asked if requirements were written, it did not ask if they were verified. |
| Requirements to Code<br>***Problem*** | PA 5.5.5<br>PA 5.5.8<br>PA 5.5.9<br>MV 4.2.9.A | There shall be complete traceability of requirements to design, code, and test.<br>Source code shall be analyzed to determine if it meets requirements.<br>Tests shall be reviewed to ensure requirements are covered.<br>IV&V shall assure that the article matches the requirements and the design specs, that all fault detection and isolation paths are tested, that all required planned actions and responses are tested. |
| | | If the intent of this requirement is to map requirements to modules, this requirement can be performed when a definition for KBS modules is determined (see Components (Parts) on page 11). If the intent of this requirement is to map individual requirements to individual rules (or other KB items), application of this requirement to KBSs will be a problem because it requires a level of detail not normally done for KBSs. This level of detail may not only be extremely difficult, but prohibitively expensive as well.<br>The survey did not contain a question on this item, although one project developed a knowledge document which, if considered as requirements, was mapped to the KB. |

| Table 2 (Page 3 of 3). Attributes of Work Products That Must Be Verified, Analyzed or Tested | |
|---|---|
| **V&V Item**<br>**Problem Status** | **Paragraph** **V&V Requirement**<br>**Analysis and Comparison** |
| Performance (Resource Usage)<br>***Problem*** | **PA 5.7** Software/hardware interface will be analyzed to ensure that "The software does not overstress the hardware."<br>**MV 4.2.3.A** Testing shall cover: External, operational, functional, performance, interface, and support requirements.<br>**MV 4.5.L** Subsystem performance evaluation shall use real flight "signals and software" to the extent possible.<br>Application of this requirement to KBSs is a problem because of the nondeterministic nature of KBSs.<br>Performance analysis was cited as a major problem in the survey. |
| Functionality<br>*Not unique* | **MV 4.2.3.A** Testing shall cover: External, operational, functional, performance, interface, and support requirements.<br>**MV 4.2.9.A** IV&V shall assure that the article matches the requirements and the design specs, that all fault detection and isolation paths are tested, that all required planned actions and responses are tested.<br>**MV 4.6.E** Prelaunch ground testing shall emphasize functional testing instead of performance testing using built-in test functions of the software.<br>There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey results indicate that functional testing was very widely utilized on existing KBSs. |
| Interfaces<br>*Not unique* | **MV 4.2.3.A** Testing shall cover: External, operational, functional, performance, interface, and support requirements.<br>Interfaces are typically between the knowledge base system shell and conventional software which is invoked as a result of some knowledge base element evaluation. There is no property of KBS interfaces that would distinguish the application of this requirement between KBSs and conventional software systems.<br>The survey results indicate that there is not a problem with interfaces between a KB and conventional programs. |
| Out of Range Values<br>*Not unique* | **MV 4.2.1.AN** Software shall be tested with out of range values.<br>**MV 4.2.3.D** Test shall cover both nominal and extreme conditions.<br>**Note:** There are several references to levels of testing, including a reference to "the four levels of testing, but no description of the four levels exists and it is suspected, based on evidence on p.4-28 and dangling references, that it has been deleted.<br>For some applications, out-of-range values may not be obvious, but otherwise there is no property of KBS interfaces that would distinguish the application of this requirement between KBSs and conventional software systems.<br>One interviewee indicated that out-of-range values were explicitly tested. |
| Human Factors<br>*Not unique* | **MV 4.11** A human factors verification program shall be established.<br>There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey did not contain a question on this item. Interviewees indicated that they were very successful in analyzing human factors considerations. |

Table 3. Items That Must Undergo V&V

| V&V Item<br>Problem Status | Paragraph V&V Requirement<br>Analysis and Comparison |
|---|---|
| Off-the-shelf (OTS) Software Components<br>*Problem* | PA 4.4.1 Procured products must be checked for adherence to Q&A requirements.<br>MV 4.4.4.A OTS software shall require full certification (including KBS shells).<br>Since KB products are not certified (unlike Ada, which is "certified" by the Ada Compiler Validation Capability (ACVC) test suite), this may be a problem.<br>The survey indicated that off-the-shelf software components (ES shells) were not subjected to explicit V&V. |
| Changes, Additions<br>*Not unique* | MV 4.2.1.H,<br>MV 4.2.1.AD If software is changed or removed, the effected system must be reverified.<br>MV 4.2.1.J Additions to an existing system must be verified and plans must be made for their checkout on-orbit prior to their activation.<br>It appears that there is no property of KBSs that would distinguish the application of on-orbit checkout between KBSs and conventional software.<br>The survey results did not indicate that there would be any problems reverifying a system after a change or addition. Sensitivity to change did not appear to be a problem. |
| Paths (Redundant Paths, Decision Paths, Executable Lines of Code)<br>*Problem* | MV 4.2.1.U Redundant software paths must be verified on the ground before they are used.<br>MV 4.2.1.AO All decision paths and executable lines of code shall be exercised.<br>MV 4.2.8.B Flight and ground software paths shall be verified prior to use.<br>MV 4.2.9.A IV&V shall assure that the article matches the requirements and the design specs, that all fault detection and isolation paths are tested, that all required planned actions and responses are tested.<br>MV 4.5.E Checkout of alternate and redundant functional paths and modes shall be required.<br>MV 4.6.A Recycled elements shall have redundant paths not utilized on previous flights tested prior during prelaunch and checkout activities.<br>Determination of paths (or rule interactions) is inherently difficult for a KBS for which no reasonable process currently exists. A workable definition of KB modules would simplify the process, but tools and/or techniques need to be developed to assist in the process.<br>60% of the survey respondents indicated that structural testing (coverage of all rules) was performed. This only indicates that a lower level of testing, corresponding to branch testing of conventional software, was performed. Most of the surveyed systems did not have complex rule interaction (which would define the "paths" in a KBS), and all persons interviewed stated that they did not test rule interactions. |
| Support Equipment<br>*Not unique* | MV 4.8 V&V for support equipment include development, certification, acceptance and functional tests.<br>MV 4.8.1.C For checkout of GSE, development test data shall be used.<br>MV 4.8.1.I All GSE will have an acceptance test to demonstrate that acceptance specifications are met.<br>MV 4.8.1.L All GSE shall be verified prior to interfacing with flight S/W.<br>There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey did not address this concern. |
| Experiment and Payload Software<br>*Not unique* | MV 4.12 Experiment and payload software must be safety verified.<br>There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey did not address this concern. |

Table 4. V&V Requirements Relating to the V&V Process

| V&V Item | Paragraph | V&V Requirement |
|---|---|---|
| **Problem Status** | **Analysis and Comparison** | |
| Reviews Must Be Conducted<br>*Not unique* | **PA 1.9** | Reviews will be conducted at specific milestones in order to assure that safety and reliability requirements are being considered. |
| | **PA 2.1.4.2** | Reviews will be conducted for KBS even if they are part of a payload. |
| | **PA 5.2.4** | Code should be inspected prior to integration and test. |
| | **PA 5.5.3** | Formal review will occur to analyze the S/W requirements derived from system requirements. |
| | **MV 4.2.9.B** | V&V activities shall include: Participation in reviews. |
| | There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey results indicate that review and inspections were done on many systems, but does not indicate how effective the review were. | |
| Q&A Must Approve Proposed Changes<br>*Issue (Life-Cycle)* | **PA 4.2.3** | Proposed changes must be reviewed by Q&A to check for quality impact. |
| | This is an issue in frequently changing systems (which expert systems seem to be). A KBS is typically developed through rapid iteration: a change is made to the knowledge base and the KBS is immediately executed to determine the effects of the change. The approval process must be accounted for in the definition of a life-cycle which also enables reasonable productivity to be retained.<br>Some of the developers surveyed felt very strongly that KBS development would be seriously hindered if rapid iteration could not be employed. | |
| Configuration Management (CM) Must Be Done<br>*Issue (Life-Cycle)* | **PA 5.2.5** | Consistency between software used and what should be used (i.e., CM) shall be done. |
| | In the state-of-the-practice, KBS development is typically performed without a defined life-cycle and without CM. The definition of the KBS life-cycle will have to account for configuration management issues.<br>Some of the developers surveyed felt very strongly that KBS development could not be performed in the presence of rigorous CM. | |
| V&V Must Be Done Throughout the Life Cycle<br>*Not a problem* | **PA 5.1.2** | V&V shall be done throughout the S/W life cycle. |
| | **MV 4.2.3.A** | Testing shall cover: Requirements compliance at several points in the life cycle. |
| | **MV 4.4.2.G** | Certification based on verifying design requirements shall be performed at each level of development. |
| | Since the life cycle mainly seems to revolve around small quickly made changes, evaluation probably is done "continually" as the system is evolving. Performance of V&V throughout the life-cycle should not be a problem once a life-cycle is defined.<br>The survey did not determine when evaluation activities were done, so it is not known whether they were all done at the end. | |
| Criticality 1 Software Must Be Verified<br>*Not unique* | **MV 4.2.1.S** | Criticality 1 software is the only software that absolutely has to be verified before launch. Others can possibly be verified during operational use; they must be approved before this can be done so. |
| | There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey did not address this concern. | |

Table 5. Methods of V&V

| V&V Item<br>Problem Status | Paragraph | V&V Requirement |
|---|---|---|
| | **Analysis and Comparison** | |
| Analysis, Test, Inspection, and Demonstration<br>*Not unique* | MV 4.2.1.A | Approved V&V methods include analysis, test, inspection, and demonstration. |
| | MV 4.2.2.A | Analytical methods shall be used along with testing. |
| | MV 4.2.3.D | Testing at each level has 3 logical parts:<br>a) Functional testing<br>b) Interface testing<br>c) Performance testing |
| | MV 4.2.4 | Inspection shall be used when applicable, software inspection will be done in the form of code, design, and security walk-through. |
| | MV 4.2.5 | Verification of software shall be done by demonstration when applicable but must be combined with other forms of verification.<br>**Note:** Demonstration means observing the software without special test hooks of any kind. |
| | MV 4.2.9.B | V&V activities shall include: Evaluation and analysis. |
| | MV 4.4.1.A | All software shall be certified for flight by analysis, test, inspection, demonstration, or a combination. |
| | MV 4.2.2.C | Analysis shall be used when flight environment can not be adequately simulated. |
| | Since most KBS development is done iteratively and checked by execution after each iteration, application of this requirement to KBSs should not be a problem. There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software systems.<br>Survey results indicated that testing was heavily used in KBS development. Many participants indicated that they did use desk checking, but the survey did not question how effective these efforts were. | |
| IV&V Must Be Performed (At Least On Critical S/W)<br>***Problem*** | PA 5.9 | IV&V shall be performed. |
| | MV 4.2.7.A | IV&V must be used on criticality 1 software (p. 4-29 is referenced). |
| | MV 4.2.9.A | IV&V shall assure that the article matches the requirements and the design specs, that all fault detection and isolation paths are tested, that all required planned actions and responses are tested. |
| | True V&V can not be done using the same expert. If different experts are used, there is typically difficulty in reconciling their views on what the system should do.<br>The survey responses indicated that IV&V was performed on about 25% of KBSs, but there was no indication of whether it was a problem. Performing truly independent V&V appears to be a problem since the survey indicated that 80% of projects relied on the expert for requirements, 60% relied on the expert for development testing, and 70% relied on the expert for system test. When multiple experts were involved, they agreed only 85% of the time. | |
| Prototyping (Instead of Analysis)<br>*Not unique* | MV 4.3.1.A | Software prototyping may be done instead of analysis at the unit level if analysis may not provide adequate assurance. |
| | There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey did not address this concern. | |
| Development Testing (Informal Testing Instead of Analysis)<br>*Not a problem* | MV 4.3.3.A | System development tests may be performed instead of analysis if it is determined that analysis can't assure that the design is adequate. |
| | MV 4.3.4.A | System development testing shall be confined to those things that can not cost effectively be tested at lower levels. |
| | MV 4.8.1.H | Certification requirements for GSE are to be met by development test whenever the criteria for certification can be met. |
| | Since most KBS development is done iteratively and checked by execution after each iteration, application of this requirement to KBSs should not be a problem.<br>The results of the survey indicate that development testing of KBSs is being performed without any problems. | |
| Endurance Testing<br>*Not unique* | MV 4.5.N | Software endurance testing shall be performed on all software during V&V, the time duration is determined on a case-by-case basis. |
| | Selection of a meaningful endurance test time may be difficult (see Endurance Test Time on page 13). There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey did not address this concern. | |

Table 6. Miscellaneous V&V Requirements

| V&V Item<br>Problem Status | Paragraph V&V Requirement<br>Analysis and Comparison |
|---|---|
| Each Level of Verification Must Ensure an Executable End-to-End System<br>*Not unique* | **MV 4.2.1.AI** Each level of verification should ensure an executable end-to-end system.<br>Currently, this does not seem to be a problem with KBSs since most systems are interpretive and allow efficient check-out in a rapid development environment. This may become a problem when a life cycle for KBS development is selected.<br>The survey did not address this concern. |
| Verification of All Software Should Be on a Uniform Set of Requirements<br>*Not unique* | **MV 4.2.1.AK** All verification should on a uniform set of requirements.<br>Since there are problems writing requirements for expert systems, the KBS requirements will probably not be at the same level as all other software, complicating application of this requirement.<br>The survey did not address this concern. |
| Analysis Methods Must Be Based on Sound Engineering Approaches<br>*Issue (Requirements)* | **MV 4.4.1.C** Analysis methods used for certification shall use "sound engineering approaches" and this should be justified by accompanying rationale.<br>Currently, there are few analysis methods for V&V of KBSs that are based on sound engineering approaches. Those that are available may be too difficult to apply. Analysis methods must be accounted for when the process by which KBS requirements gathered and recorded are defined.<br>The survey did not address this concern, though methods used, such as structural testing, seemed to be ad hoc. |
| Certification Must Be Done on Integrated System Regardless of the Classification of the Parts<br>*Not unique* | **MV 4.4.2.H** Certification of flight software must be performed on the integrated system regardless of the classification of the individual parts.<br>This requirement applies to two different levels:<br>1. A KBS has to be certified regardless of the level of certification of each of its modules.<br>2. The subsystem which the KBS is part of must be certified even if the KBS and all other subsystem components have been certified.<br>There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey did not address this concern. |
| Onboard Software Built-in Test, Checkout, Monitoring, and Isolation Capabilities<br>*Not unique* | **MV 4.6.D** Onboard flight software capabilities shall be used to the maximum extent possible in order to minimize special ground test software.<br>**MV 4.7.2.D** Onboard automatic checkout capability, health monitoring, and fault isolation shall be used for test activities (on-orbit).<br>This is difficult for conventional software systems and probably has not been attempted for KBSs. There is no property of KBSs that would distinguish the application of this requirement between KBSs and conventional software.<br>The survey did not address this concern. |